python for Computational Problem Solving - pCPS - Dictionaries_Sets Lecture Slides - Class #19_#20

Nitin V Pujari
Faculty, Computer Science
Dean - IQAC, PES University

pCPS Assignment Batches

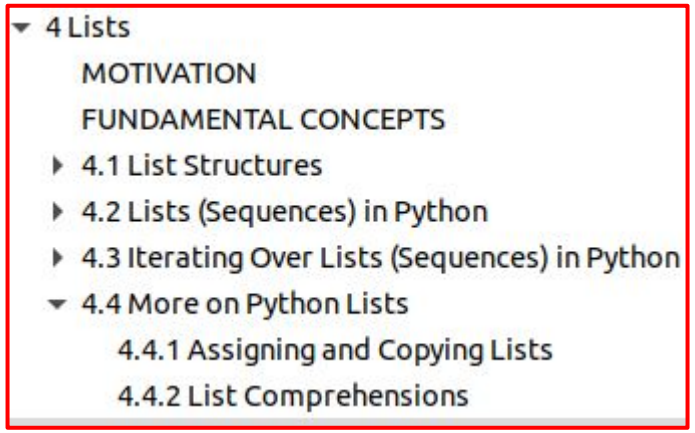
```
,BatchId,ProjectBatch
0,pCPS_Assignment_Batch_ID_1,"('PES1202100893', 'PES1202100956', 'PES1202101345')"
1,pCPS_Assignment_Batch_ID_2,"('PES1202100862', 'PES1202101351', 'PES1202100999')"
2,pCPS_Assignment_Batch_ID_3,"('PES1202100802', 'PES1202100895', 'PES1202101314')"
3,pCPS_Assignment_Batch_ID_4,"('PES1202101342', 'PES2202100686', 'PES2202100705 ')"
4,pCPS_Assignment_Batch_ID_5,"('PES1202100868', 'PES1202100891', 'PES1202101354')"
5,pCPS_Assignment_Batch_ID_6,"('PES1202100884', 'PES1202100886', 'PES1202101033')"
6,pCPS_Assignment_Batch_ID_7,"('PES1202101027', 'PES1202101339', 'PES1202101054')"
7,pCPS_Assignment_Batch_ID_8,"('PES1202100959', 'PES1202100991', 'PES1202101048')"
8,pCPS_Assignment_Batch_ID_9,"('PES1202101466', 'PES1202101481', 'PES1202100838')"
9,pCPS_Assignment_Batch_ID_10,"('PES1202101050', 'PES1202101415', 'PES1202100970')"
10,pCPS_Assignment_Batch_ID_11,"('PES1202100960', 'PES1202100860', 'PES1202100967')"
11,pCPS_Assignment_Batch_ID_12,"('PES1202100974', 'PES1202100877', 'PES1202101330')"
12,pCPS_Assignment_Batch_ID_13,"('PES1202100801', 'PES1202101349', 'PES1202101480')"
13,pCPS_Assignment_Batch_ID_14,"('PES1202100803', 'PES1202101020', 'PES1202101513')"
14,pCPS_Assignment_Batch_ID_15,"('PES1202101315', 'PES1202101458', 'PES1202101460')"
15,pCPS_Assignment_Batch_ID_16,"('PES2202100680', 'PES1202100836', 'PES1202101014')"
16,pCPS_Assignment_Batch_ID_17,"('PES2202100695', 'PES1202101416', 'PES1202100930')"
17,pCPS_Assignment_Batch_ID_18,"('PES1202100816', 'PES1202101407', 'PES1202100890')"
18,pCPS_Assignment_Batch_ID_19,"('PES1202100829', 'PES1202101353', 'PES1202100841')"
19,pCPS_Assignment_Batch_ID_20,"('PES1202100789', 'PES1202101306', 'PES1202100830')"
20,pCPS_Assignment_Batch_ID_21,"('PES1202101329', 'PES1202100807', 'PES1202101038')"
21,pCPS_Assignment_Batch_ID_22,"('PES1202101041', 'PES1202100835', 'PES1202101051 ')"
22,pCPS_Assignment_Batch_ID_23,"('PES2202100627', 'PES1202100864', 'PES1202101358')"
23,pCPS_Assignment_Batch_ID_24,"('PES1202100928', 'PES1202101522', 'PES1202100953')"
24.pCPS_Assignment_Batch_ID_25,"('PES1202101538', 'PES1202101325')"
```

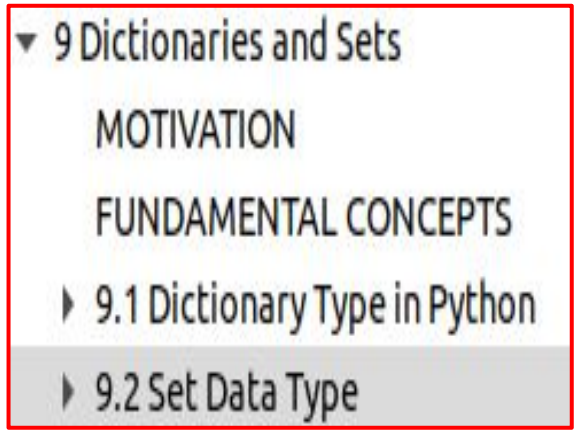
python for Computational Problem Solving Syllabus

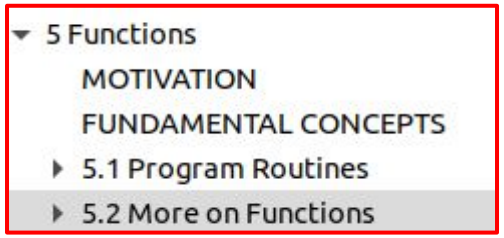
Unit II: Collections & Basics of Functions - 12 Hours

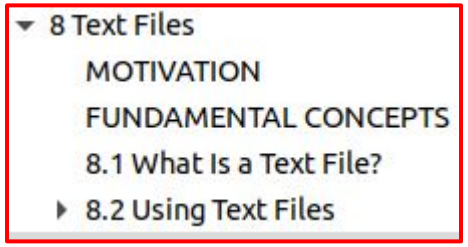
Lists, Tuples , Dictionaries, Sets, Strings and text file manipulation: reading and writing files. Functions : Definition, call.

T1: 4.1 – 4.4 - Class #15, #16, #17, #18
T1: 9.1 – 9.2 - Class #19, #20, #21, #22
T1: 5.1-5.2 - Class #23, #24
T1: 8.1, 8.2, 8.3 - Class #25, #26

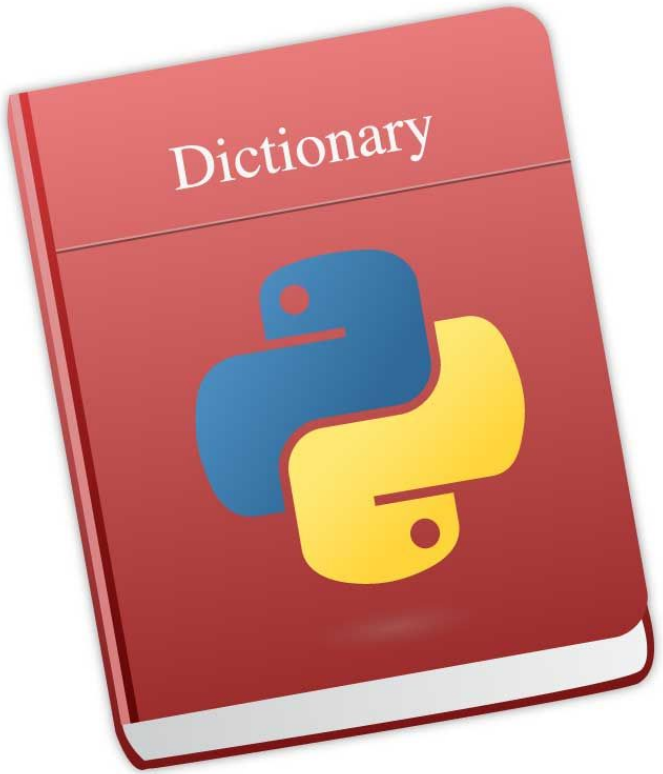








pCPS 9 python Dictionary



pCPS 9.1 Dictionary Type in python

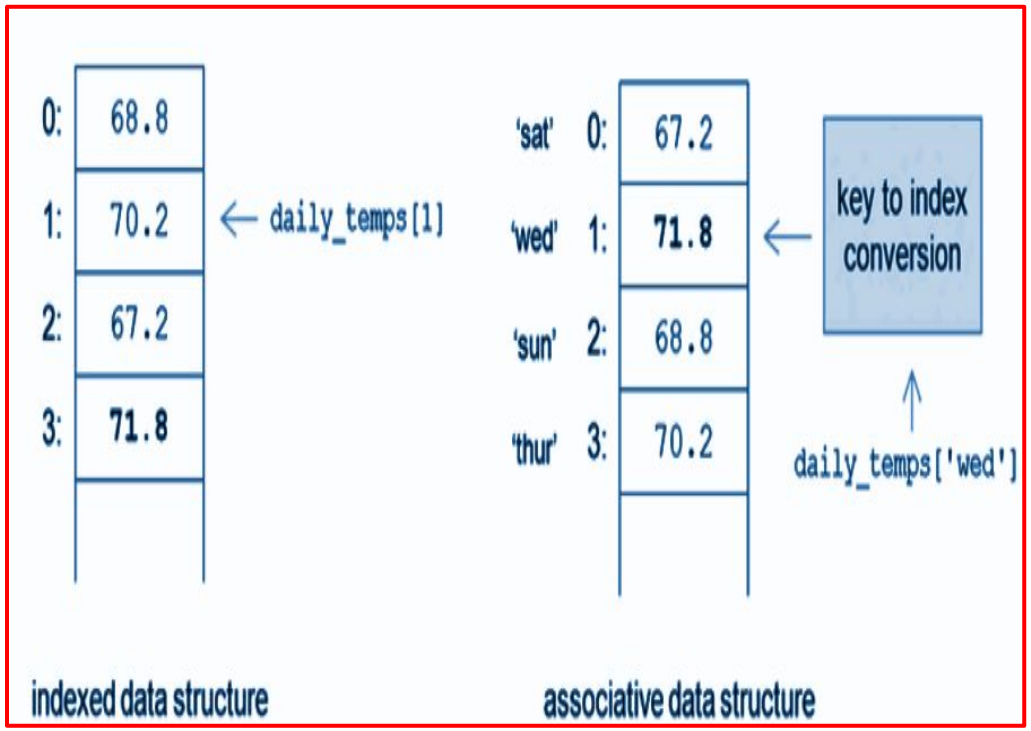
- In this topic we introduce the notion of an **associative data structure**

- The elements of an associative data structure are **unordered**, instead **accessed** by an associated **key** value

- In python, an associative data structure is provided by the **dictionary type** .

pCPS 9.1.1 Dictionary in python

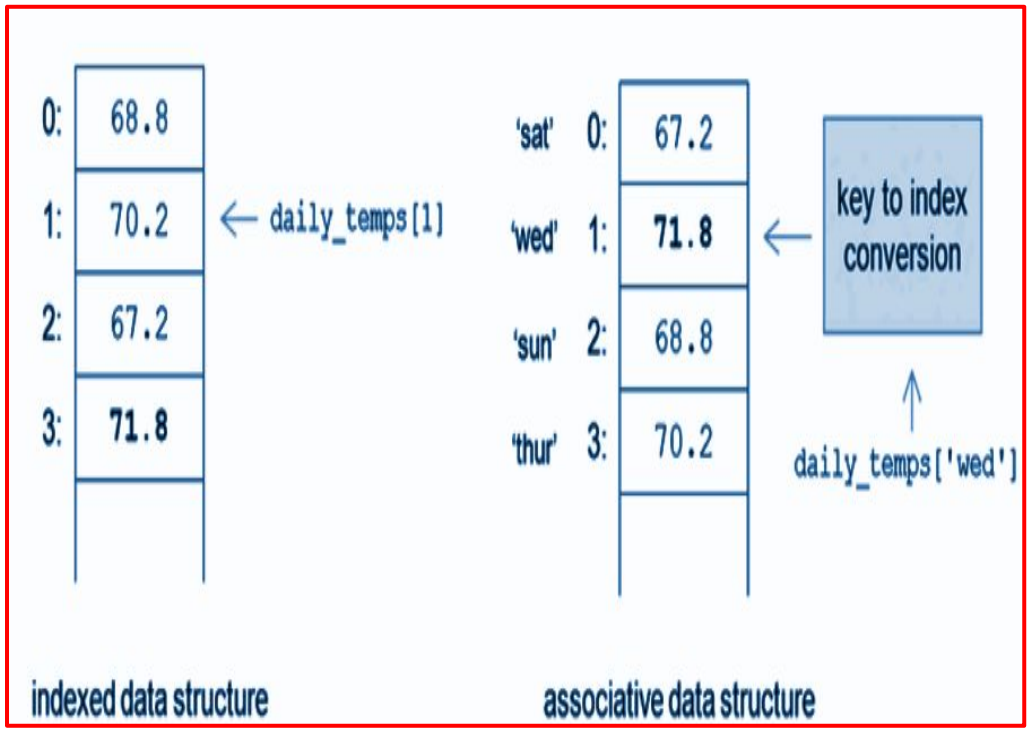
- A **dictionary** is a **mutable**, associative data structure of **variable length**.
- On the left is an indexed data structure, and on the right an associative data structure.
- **Even though** the elements of the associative data structure are **physically ordered**, the **ordering** is **irrelevant** to the **way** that the **structure** is **utilized**.



pCPS 9.1.1 Dictionary in python

- The **location** that an element is stored in and retrieved from within an associative data structure **depends** only on its **key** value , thus there is **no** logical **first** element, **second** element, and so on.

- The **specific location** that a value is stored is determined by a particular method of converting **key values** into **index values** called **hashing**

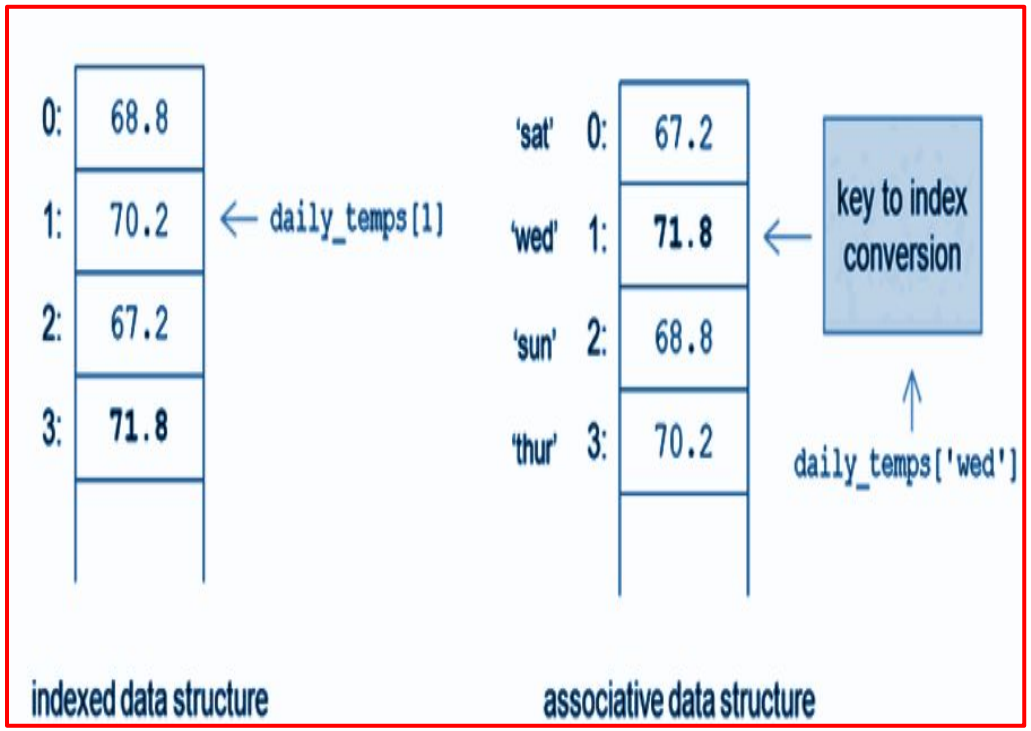


pCPS 9.1.1 Hashable in python

- An **object** is **hashable** if it has a **hash value** which **never changes** during its **lifetime**

- **Hashability** makes an object **usable** as a **dictionary key** and a **set member**, because these data structures **use** the **hash value internally**

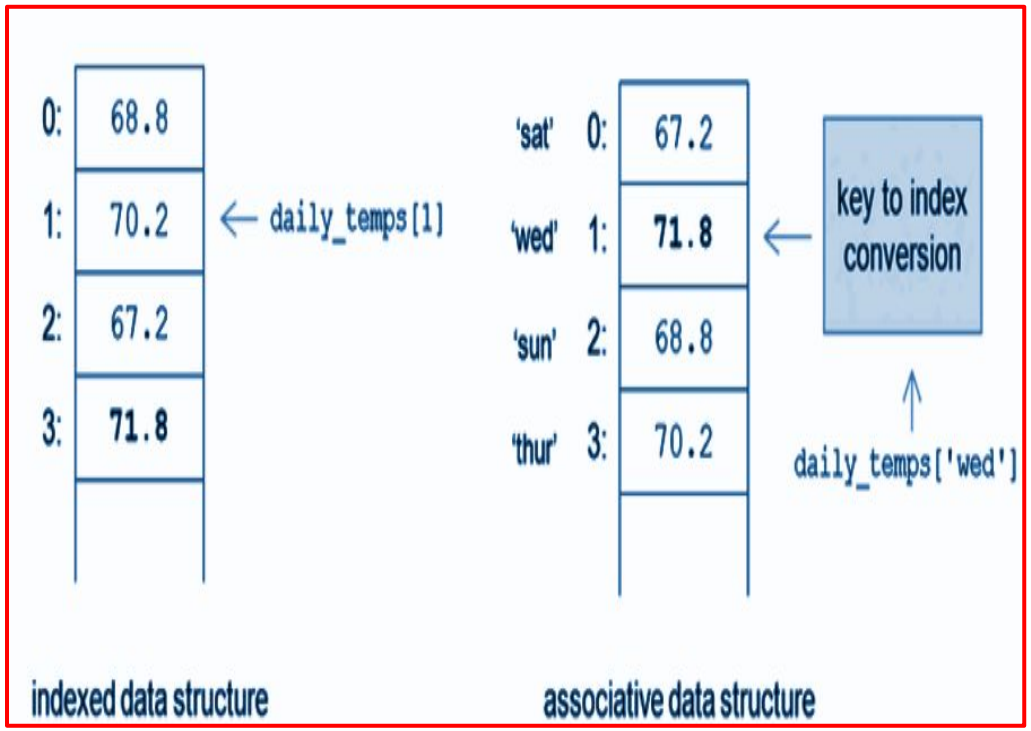
- All of python's **immutable built-in** objects are **hashable**, while **mutable containers** (such as lists or dictionaries) are **not hashable** .



pCPS 9.1.1 Hashable in python

- **Objects** which are **instances** of **user-defined classes** are **hashable** by **default**; they all compare unequal, and their hash value is their id()

- **Hashing** is a **concept** in computer science which is used to create high performance, **pseudo random** access data structures where **large** amount of **data** is to be **stored** and **accessed quickly**



pCPS 9.1.1 Operations on Dictionaries in python

Operation	Results
`dict()`	Creates a new, empty dictionary

`dict(s)`	Creates a new dictionary with key values and their associated values from sequence `s`, for example,
	`fruit_prices = dict(fruit_data)`
	where `fruit_data` is (possibly read from a file):
	`[['apples', .66],…,['bananas', .49]]`

`len(d)`	Length (num of key/value pairs) of dictionary `d`.
`d[key] = value`	Sets the associated value for `key` to `value`, used to either add a new key/value pair, or replace the value of an existing key/value pair.
`del d[key]`	Remove key and associated value from dictionary `d`.
`key in d`	`True` if key value `key` exists in dictionary `d`, otherwise returns `False`.

THANK YOU

Nitin V Pujari
Faculty, Computer Science
Dean - IQAC, PES University
nitin.pujari@pes.edu

For Course Digital Deliverables visit www.pesuacademy.com